Clustering

Wednesday, September 22, 2021 4:34 PM

Clustering = grouping/assignment of data points such that objects in the same cluster are similar and dissimilar to objects in other groups

Two Main Use Cases for Clustering

Outlier detections / Anomaly Detection

- Data cleaning/processing
- Credit card fraud, spam filter, etc.

Filling gaps in your data

Clusters can be ambiguous

Types of Clusterings

Partitional

Each object belongs to exactly one cluster

Given **n** data points and a number **k** of clusters, partition the **n** points into **k** clusters

Take a given cluster, compute distances between points assigned to the cluster, repeat for each cluster, sum up all the cluster totals. If the total is small, then the partitioning is good

K-means - Lloyd's Algorithm

1. Randomly pick **k** centers $\{u_1, \ldots, u_k\}$
2. Assign each point in the dataset to its closest center
3. Compute the new centers as the means of each cluster
4. Repeat 2 & 3 until convergence

Final convergence is based upon the randomly picked centers in step 1, therefore it does not always give an optimal solution

Starting with centers too close to each other may be problematic

Picking points with the farthest distance is susceptible to outliers (Farthest first traversal)

K-means++

1. Start with a random center
2. Let D(x) be the distance between **x** and the centers selected so far. Choose the next center with probability proportional to $D(x)^a$

When a=0, random initialization

When a=infinity, farthest first traversal

When a=2, K-means++

K-means++ always better than k-means

K-means limitations:

Does not handle varying densities very well

Often times splits up larger clusters

Performs poorly with clusters that are non-globular

How to choose the right **k**?

1. Iterate through different values of **k** (elbow method)
2. Use empirical / domain-specific knowledge

K-means variations

1. K-medians (uses the L_1 norm / manhatten distance)
2. K-medoids (any distance function + centers must be in the dataset
3. Weighted K-means (each point has different weight when computing the mean)

Hierarchical

A set of nested clusters organized in a tree

Density-Based

Defined based on the local density of points

Soft Clustering

Each point is assigned to every cluster with a certain probability